

From: C.K. Haun [REDACTED]
To: Matt Fischer [REDACTED]
CC: Bill Hudson [REDACTED], Garrett Rice [REDACTED]
BCC:
Subject: Fwd: Why Adobe doesn't participate more in the Mac Apps Store
Attachments:
Sent: 02/11/2014 06:44:17 PM 0000 (GMT)

Met with Adobe about their iOS work the other week, and asked them about their limited participation in the Mac App Store. They provided a rather comprehensive list of items, many of them related to the business side of the Mac App Store. Food for thought.

—ck

Begin forwarded message:

Hi C..K. and Bryan,

In our last meeting with you and Shaan – you had asked me why Adobe didn't participate more in the Mac Apps Store. To respond to you completely, Dave and I compiled a list of reasons we gathered from multiple product management and engineering staff which we assembled below. This is just a compiled list -- the rigor of the tone isn't meant to come off as profusely complaining ;-)

If you have questions about our rationale for preferring what isn't supported, we are happy to go back to our teams and get more data.

Thanks!
Ginger

Summary:

The Mac App Store charges more than our current channels (30% domestic, ~50% foreign), has very expensive terms/costs, has less sell-thru, and adds many barriers between Adobe and its customers.

Details:

- Pricing
 - Other channels allow us more flexibility by Geo, we can put taxes on top (instead of making them inclusive and creating channel conflict)
 - App Store doesn't allow upgrades, coupling/bundling/suites, complex business models (hybrid, subscriptions, etc.); control over education pricing (either 50% off or nothing, and they allow people to be "students" for up to 2 years after school); Free trials are restricted, and Enterprise requires custom builds and a complex sign-up system
- Channel
 - As a bigger vendor we normally get to leverage our size/importance to get better placement, presentation, promotion and we can do bundling — while the Mac app store requirements favor smaller/simpler apps. The playing field is leveled towards or advantages smaller developers (competitors can easily enter the market and gain visibility, and their Apps don't require expensive re-engineering to comply with terms).
- Customer Relationship
 - We have no control of promotions (in store), and poor communications on whether we'll be featured or how (so we can't coordinate efforts, only delay releases and hope)
 - We get little to no analytics on who our customers are, even abstracted, unless they opt-in
 - It's very hard to time releases because we don't know how long reviews will take. Best case it is days, worst case it is months and requires engineering rework (terms change over time).
 - Our positioning is based on customer reviews, but we have no mechanisms to respond to bad reviews or

**Exhibit
0133**

PX-0133.1

customer complaints. (Poor feedback channels)

- Mac App Store is an island — we can't promote other soft goods within the store or within our Apps on the store (treated like channel conflict, and a violation of terms).
- No control over refunds or remediation if there is a problem
- We can't do validation of how many units an App is installed on, which makes licensing restrictions hard (we have to pay royalties on a per unit/installed basis or re-negotiate contracts with patent pools and large organizations)
- Technical
 - Applications have to be unified into a single payload (no companion/helper apps), requires expensive sandboxing
 - Shared files, shared messaging, and many other things that make suites of Apps work better, are blocked or expensive in this model — wasteful disk space and bad customer experiences
 - Not only are there huge up-front costs with re-architecture, but it requires de-featuring apps. Once customers recognize that there's less functionality in the Mac App Store versions, our reviews get hammered (so far we've never had a positive review for functionality we were forced to remove).
 - Apple will change restrictions on Apps (like turning off deprecated API's mid cycle) without warning, blocking updates — resulting in huge support costs
 - Limited version control: we can't control whether customers can download older versions. If we fork a product (drop support for old OS or hardware) we can longer update the older version (even security fixes)
 - No betas are allowed
 - Very difficult to plan software because lots of terms are vague, so we don't know until after App Review accepts how much work there is to do. Could be good, or could be months of additional effort.
- Who is the channel?
 - App Store customer base appears to be more often impulse buyers (looking at low cost products), and not as many pro/aspiring customers willing to put in longer training efforts to learn pro-workflows — so our capture rates are lower than some other channels, and the customer frustration is higher. Our reviews suffer because of that customer-base.
 - Really this channel is better for entry products that are easier to learn and use, but not as powerful — but since this channel is locked in by terms, we can't do things that might make it more valuable — like creating entry level versions of our products, and being able to move customers up to other pro-products later.
 - Since we have no control over who the customer is, we can't do upgrades out of the channel (take them from an in-channel entry version to a more full featured Pro version).

--CKH